Cooperative Learning Model based on Multi-Agent Architecture for Embedded Intelligent Systems

Mónica Villaverde, David Pérez and Félix Moreno

I. INTRODUCTION

Artificial Intelligence (AI) is a multidisciplinary science that includes several aspects from informatics, philosophy, human reasoning, and biological behaviors. In this area the most significant contributions were produced from the forties, nevertheless the evolution of AI has been slower than expected. In 1950 the *Turing Test* -developed and published by Alan Turing [1]- supposed a turning point in AI evolution. However, the term of AI was not coined until 1956, in the Dartmouth College where John McCarthy and other experts shared knowledge and ideas about what AI should be. Years after this meeting, the first genetic algorithms were introduced and the automatic evolution was founded. Subsequently, between the early eighties and the early nineties, the first commercial expert system was developed and the concepts of data mining and intelligent agents were also introduced. Moreover, new ideas have been developed as Swam Intelligence or Ambient Intelligence (AmI). Nowadays, the AI has improved a lot and there are many systems that include many of its aspects.

Cooperation is a useful tool to enhance the behavior of an AI system by combining it with other intelligent mechanisms. One of its main strengths is that a cooperative system fits properly into the topic of swarm intelligence. It can be used to control and to combine several agents which act as single entities inside the swarm community. However, there are other alternatives appropriated to use cooperation, such as multi-agent systems or blackboard-based systems [2] [3]. In this work a cooperative implementation is used to improve the behavior of an identification system creating a more reliable one. Employing a cooperative model, a communication support is also required. Hence, a wireless network has been implemented in order to provide an infrastructure for data transfer among the devices that compose the network. Furthermore, using an appropriated methodology it is also possible that the system acquires learning capabilities. In this paper, in order to analyze different ways to tackle the same problem using learning methodologies two weighting procedures are proposed. Besides, to compare its results with the previous ones a non-learning method is also described. One of the main advantages of the presented approach is that it can reduce the influence of the environment. Therefore, the new system is more versatile, reliable and can be used in different scenarios. This makes that the topics discussed in this paper can also be viewed within the Ambient Intelligence (AmI) issue due to the fact that the proposed system senses the environment, analyzes the obtained data, reasons about them and acts over the environment again [4]. Consequently, the system has a strong link with the environment model.

The rest of the paper is organized as follows. Section II presents the related work about cooperative systems combined with different AI procedures. The proposed cooperative model and its functionality are explained in section III whereas section IV specifies its voting mechanism and its learning capabilities. Finally, the results and the conclusions for each voting procedure are given in section V and VI respectively.

II. RELATED WORK

Nowadays, intelligent cooperative systems are used for multiple activities. Many researches address the issue of the implementation of this kind of systems. In [5] a distributed swarm of robots is introduced. Robots cooperate to explore dangerous areas or inaccessible territories to seek targets. In that case, the use of swarm intelligence provides scalability and robustness. In that approach a distributed topology is employed whereas in this paper a centralized scheme is proposed to enhance the behavior of an identification system. This kind of topology is also used in [6] where the leader of the centralized topology obtains a final data detection by using the maximum likelihood criteria in a multiple-input multiple-output (MIMO) broadcasting system. Moreover, Bobtsov *et al.* [7] propose a multi-agent system of aerial vehicles. Their approach consists of creating a system to monitor ecological areas of difficult access. Each aerial vehicle is an agent in the multi-agent system and each one is in charge of executing a different part of the entire mission. Any agent cooperates with the other ones by sharing the execution of a specific and particular task. However, they need each other to complete the entire mission since each task of an agent has influence over the other ones. Therefore a kind of cooperation is required. Differing from this, our approach tackles a multi-agent system where every physical device is able to generate its own individual identification and also implements more than one agent since each device. Hence, in this paper cooperation is useful to pool all of these individual identifications.

Despite of those applications, cooperation is vastly common in the management network field where coordination plays an important role [8]. Therefore, swarm intelligence is very used to improve the network performance. In [9] authors provide an energy saving routing algorithm inspired in swarm intelligence whereas in [10] an adaptive dynamic routing algorithm is introduced using the same kind of intelligence to find the optimal routes.

III. COOPERATIVE MODEL

The main function of cooperative models is to create more powerful systems by using information from different subsystems. These kinds of models can be applied to distribute several tasks in order to reduce the system complexity. Moreover, they can also be used as an alternative to enhance system reliability when several devices are working together, reducing the unawareness of the system and giving better solutions. However, in other applications a combination of both alternatives can be more appropriate. In any case, a multi-agent architecture fits into a cooperative model since each agent is in charge of executing a simple task to later join all the information again in order to find the final solution.

This paper is focused on improving the system behavior using a cooperative model to increase the success rate (hit rate) of an identification system (Fig. 1). In these terms, cooperation is the mechanism used to know all the identification results in order to determine the best object category.

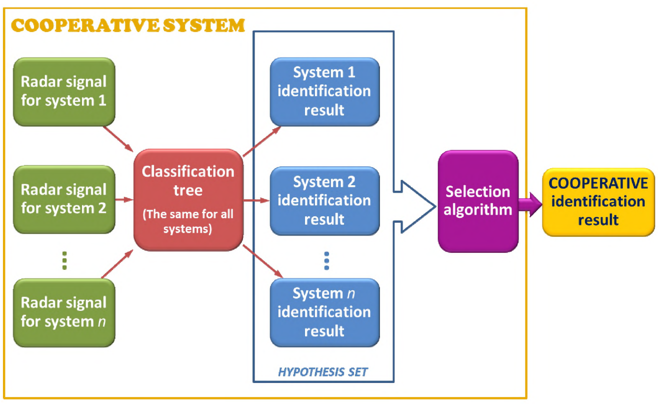

Fig. 1. Cooperative process.

The proposed process requires a communication infrastructure to transmit the information about each identification result. Hence a wireless network has also been implemented. In addition, the whole system management (including identification process and network control) is carried out by using a multi-agent architecture.

A. Background

Early implementation of this project was a low-cost identification system able to detect and identify objects along the road to control the public lighting, whose main goal was turning streetlamps on only when a pedestrian or a vehicle was on the road in order to reduce energy consumptions [11].

The system was based on a DSP (Digital Signal Processor) platform and a simple radar device (Fig. 2). The radar was in charge of detecting movements and its signal was processed by the DSP in order to determine the object category.

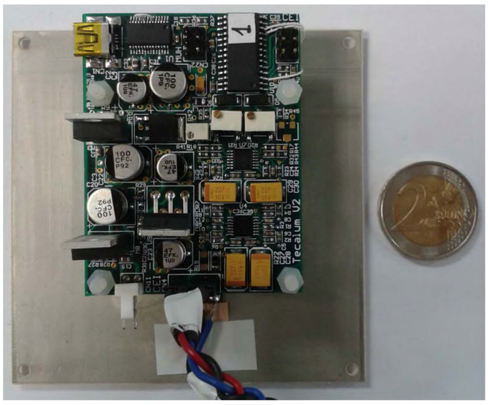

Fig. 2. Prototype of DSP-based device.

The identification process was carried out by using an expert system based on a classification tree. A classification tree -or a decision tree- is a machine learning methodology which is considered as a supervised learning technique. Therefore, the aim of the implemented classification tree is to determine the category of the object that crosses the road. For this purpose it uses a training set that contains patterns based on past experiences.

Practical results demonstrated that the location and the orientation of the radar device have a high influence on the results. Therefore a more suitable radar position is required in order to obtain successful results. Table I shows the hit rate for different real experiments.

TABLE I. SUCCESS RATE: PRACTICAL RESULTS

	Signals taken at 4-5 meters		
	Alg.3	***Alg.5***	***Alg.6***
Car	*60,00%*	*80,00%*	*60,00%*
Person	*87,72%*	*82,45%*	*81,08%*
Environment	-	-	*95,83%*

All of them were carried out with the radar located at 4-5 meters over the street level and are based on different classification tree algorithms. Algorithm 5 uses a CHAIDbased classification tree [12] whereas algorithms 3 and 6 use a CRT-based classification [13]. The difference between algorithms 3 and 6 is that the last one includes several environmental signals in its training set to considerer weather conditions. In these terms some rain patterns were included in algorithm number 6. Results demonstrate that a good training set is required to obtain high success rates. Moreover, the radar location plays an important role and investing additional efforts in this area these hit rates could be increased. This paper tries to enhance those first results taking into account different opinions from several radar devices. The fact that a radar device can be located at a good place is dependent of many factors -included the trajectory of the object-. Therefore, a network is proposed as a communication support among radar devices in order to reduce the location influence.

B. Multiagent architecture

An intelligent agent is a virtual entity that needs external information to act accordingly in order to achieve its predetermined purpose. Its objective can be reached by reasoning, using more or less autonomy and even through data transfer. Employing several agents a multi-agent architecture is created where each agent is in charge of implementing a simple or medium complexity task. Additionally, a communication mechanism has to be created in order to transfer information among agents.

In this paper, four different kinds of agents are defined. Two of them are focused on network management and the other ones are dedicated to the identification process. On the one hand, in terms of network control, the *network management agent* is in charge of creating and controlling the wireless network and applying for data from all the radar devices whereas data requests are handled through *network agents*. On the other hand, from the point of view of the identification process, the *identification agent* is in charge of providing each radar device with the identification result whereas the information collection is done by the *evaluator agent* which is also responsible for determining the final identification outcome.

C. Radar Devices Network

In this work the communication task among radar devices is carried out by using a wireless network for data transfer. Wireless technology provides better behavior regarding to system scalability and the location of each node that forms the network. Our approach uses ZigBee technology as a communication protocol since it is a flexible, low-cost and low-power standard for wireless networks. In this application, not too large data transfers are required. Therefore ZigBee is used as an appropriated solution in order to develop a simple platform for device communications.

However, despite of the advantages of wireless communications, there are other considerations that have to be analyzed. The most important one is the node coverage. In wireless sensor network the communication among nodes has to be ensured avoiding isolated nodes. To determine the node coverage the Friis formula is analyzed (1). It specifies the amount of received power and shows its dependence of several parameters as the distance D between nodes, the λ wavelength, the transmitted power P_T, the antennas' gain -both in the transmission G_T and in the reception G_R-, the losses of the links like the transmission and reception losses -L_T and L_R respectively- and also the obstacles losses, L_{OBS}.

$$\frac{P_R}{P_T} = \left(\frac{\lambda}{4 \cdot \pi \cdot D}\right)^2 \cdot G_T \cdot G_R \cdot L_T \cdot L_R \cdot L_{OBS} \qquad (1)$$

The L_{OBS} has been estimated according with the ITU (International Telecommunication Union) recommendations and considering the more extended values in other applications. Therefore, in this work, the considered value is -30 dB for indoor scenarios.

D. Cooperative system architecture

The proposed system is based on a centralized architecture, in terms of functionality, in which each node is composed by a ZigBee transceiver, a radar device as a sensor and an additional hardware platform for processing the radar signal. Each node affords its own identification result through its *identification agent*. Additionally, the coordinator node has a dual role. On the one hand it provides its own outcome but on the other hand it is in charge of determining the final identification results according to the decision of the rest of the nodes by means of the *evaluator agent*. Furthermore, the network's creation is carried out by the *network management agent* and the data transfers are done by the *network agents* through the wireless links. Fig. 3 shows the topology and specifies where each agent is implemented. Although all general nodes have to communicate with the coordinator node, data transfer between them could be done through another general node using the routing protocol included in the ZigBee stack. Once the network is created, all nodes belong to such network. In case that the coordinator falls down, the system may continue operating as long as any general node incorporates the role of the *evaluator agent* to support the system's functionality. In that situation, every partial identification result will be sent to that node that acts as the coordinator node (sink) to obtain the final identification result.

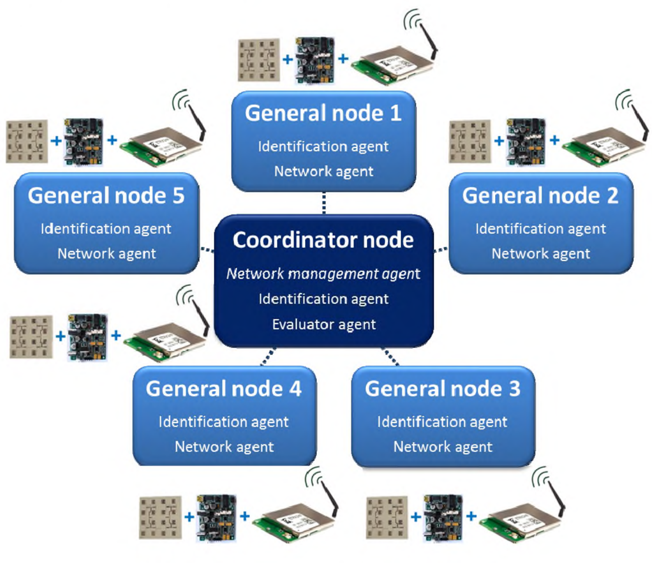

Fig. 3. Cooperative system architecture.

IV. Voting Decision and Learning

The search for a cooperative solution, done by the *evaluator agent*, can be accomplished by using different voting procedures. In this paper three alternatives are proposed in order to analyze the system behavior for each case and to find the best alternative to provide better success rates. On the one hand a non-learning methodology based on majority voting is analyzed and on the other hand two different weight-based voting alternatives with learning capabilities are proposed.

Learning is acquired by using a combination of two artificial intelligence mechanisms which fit perfectly into the idea of the proposed cooperative system: swarm intelligence and weighting assignment. Swarm intelligence is based on the concept of cooperation since several individuals work together to achieve a common goal by sharing data and following the behavior of the whole group. Therefore, all of the wireless network nodes belong to the same community and each one behaves as an individual of this community. Their common objective is to determine the category of the detected object with more reliability. Furthermore, a weighting technique is used to weight all the contributions in order to determine the final outcome based on past experiences. Each weight is updated for each detection and can be increased or decreased depending on whether the node has contributed positively or not to the cooperative result. A positive contribution is understood as a hit which means that the result of the node matches with the cooperative result.

A. Majority-based voting methodology

Majority-based voting technique does not afford any type of learning due to the fact that it does not use any mechanism for considering past experiences. The result only depends on the current partial result of each node. Hence, there are no procedures that remember the previous behavior of a particular one. This methodology consists of analyzing partial results of every node in order to determine the cooperative result taking into account the most voted category. The process to obtain the type of the detected object using majority-based methodology is shown in Fig. 4. The coordinator node is in charge of assessing the solution set and its own identification result in order to determine the most voted category.

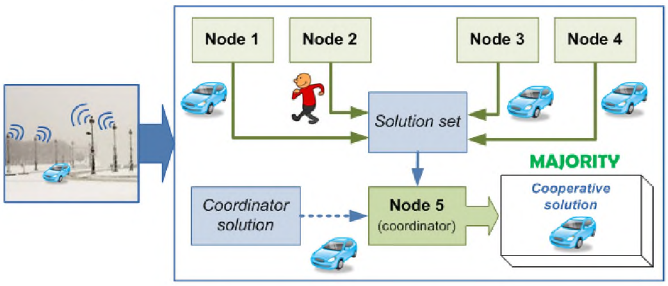

Fig. 4. Majority-based voting methodology.

This kind of voting technique has a high probability of draw occurrences. Therefore, in case of a draw, the system is not able to determine the category considering only the votes coming from the nodes. For this reason, a more complex technique based on weighting is dealt to regard the previous behaviors of the nodes for enhancing the reliability of entire system.

B. Weight-based voting methodologies

A weight-based voting technique implies that not all contributions have the same influence on the final cooperative result. Generally, weighting allows to define the impact of each partial identification. Therefore each node calculates its own result and the coordinator node has to weight this result by the corresponding value. The general procedure for weighted voting is presented in Fig. 5.

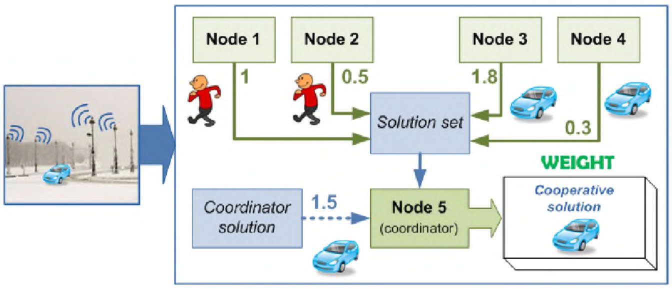

Fig. 5. General weight-based voting methodology.

Weights are used to report on the previous behavior of every node; hence they are useful to take into account past experiences. The simplest way to implement the weighting procedure is to use a *global weighting-based voting* which consists of assigning just a weight per node. In that case, weights provide information on how good each node is independently of the category of the detected object. For every object detection, the node will be rewarded if its identification matches with the cooperation result, otherwise it will be penalized. The reward consists of incrementing its weight whereas the penalty consists of decrementing it. The increment or the decrement is determined by the learning rate that indicates how much the weights between two evolutions (two detections) should be amended.

Moreover, to consider different behaviors of every node depending on the type of the detected object, *conditioned weighting-based voting* is also developed. The procedure does not manage only one weight per node since it considers the type of the detected object and the category determined by the node as its partial identification. The number of weights depends on the number of the defined categories. When three categories are established, nine different weights are defined (3x3 matrix). A general matrix for weights definitions is shown in (2).

$$\text{Matrix of conditioned weights} = \begin{bmatrix} \alpha_{Cn} & \alpha_{C/Pn} & \alpha_{C/En} \\ \alpha_{P/Cn} & \alpha_{Pn} & \alpha_{P/En} \\ \alpha_{E/Cn} & \alpha_{E/Pn} & \alpha_{En} \end{bmatrix} \quad (2)$$

The main diagonal is composed of α_{Cn}, α_{Pn} and α_{En}. These values correspond to the weight of car given car, person given person or environment given environment respectively for node *n*. The rest of the weights are related to the cases when the cooperative solution does not match the solution of node *n* (its partial identification). For example, $\alpha_{P/Cn}$ is the weight associated to the case where the identification result of node *n* is car but the cooperation result determines that the detected object is a person.

Using the *conditioned weighting-based voting,* and in order to reward or penalize each node, a more complex mechanism has to be implemented. Unlike the previous procedure, for each evolution and considering the learning rate value, more than one weight is modified instead of only one of them to adjust weights for the next object detection. In case that the node matches the cooperative result, as a car, α_C has to be increased and $\alpha_{P/Cn}$ and $\alpha_{E/Cn}$ have to be decreased. However, α_P will be decreased whereas $\alpha_{C/Pn}$ will be increased when the node determines that the detected object is a person whereas the overall system by cooperation concludes that the object is a car.

In general, any kind of weighting procedure should improve the reliability of the system versus the use of majority voting mechanism since the first ones reduce their unawareness, decreasing the number of draws, by creating a more varied solution set. In next section, to analyze the three mentioned methodologies, different tests are described.

V. RESULTS

In order to validate the previous statements, in this section the obtained results are shown. First of all, to determine the maximum distances allowed between the coordinator node and the rest of the nodes with the aim of maintaining the communication, a coverage study is presented. Then, several situations are tested to verify the functionality of the three different voting procedures mentioned in previous section.

A. Coverage analysis

By using the Friss formula (1) it is possible to analyze the maximum distances between the coordinator node and the rest of the nodes testing different scenarios with different features. In order to determine those maximum distances a sensitivity specification of ZigBee module is required. The used module specifies that its sensitivity is around -97 dBm, therefore if the received power is less than that value, the receptor module will not aware of the presence of a received signal. Considering the emitted power, the sensitivity value, the wavelength of the radar signal, the gain of the antenna and all the different losses (including the estimated L_{OBS} at -30dB for indoor scenarios), different estimations may be represented as shown in Fig. 6.

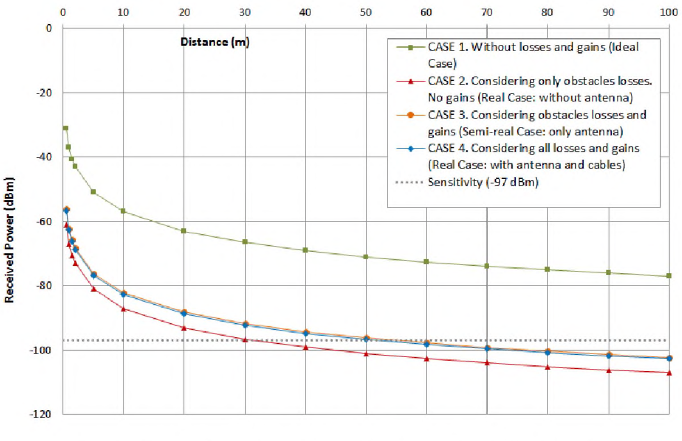

Fig. 6. Estimation of maximum distances using Friis formula.

The figure shows four different cases that demonstrate the influence of gains and losses in the received power. A more detail zoom is shown in Fig. 7 to visualize more accurately the differences between cases 3 and 4.

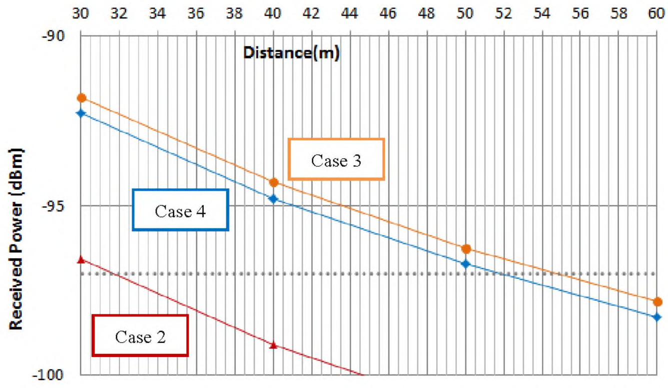

Fig. 7. Zoom of maximum distances estimations.

Results demonstrate that the maximum distance can reach 50 meters approximately for the real case (case 4), but in spite of this the recommended maximum distance for indoor scenarios should be just 30-35 meters to guarantee the communication among nodes since the analysis uses estimated values. Wireless communication was tested being demonstrated that these calculated distances are valid for the used ZigBee module.

B. Voting procedures verification

In order to verify the voting procedures mentioned in previous section, three different simulations have been carried out. Majority-based procedure is called algorithm number 1 whereas algorithms 2 and 3 are used for implementing the global and the conditioned weighting-based procedures respectively.

The first analysis shows the number of successes (or hits), misses and draws for different network sizes and its results are shown in Fig. 8. The results correspond to nodes with a success rate ranging between 40% and 90%. In case of using minor hit rates for each node, a larger network has to be created to achieve the same level of success. Moreover, if these rates are higher, a large number of hits are obtained by using small networks but there will not be a significant improvement if the network increases.

The second simulation tries to analyze how the hit rate is varied according as the system evolves. Graphs shown in Fig. 9 represent the success rate versus the number of evolutions for three different network sizes. Furthermore, Table II sums the mean and the slope up for each case in order to realize the improvement of every algorithm considering the network size. For that analysis, nodes with a success rate ranging between 40% and 90% are employed. If nodes have lower hit rates, the mean will decrease in every case. However, if the network is composed by nodes with higher success rates, also all slopes will be lower.

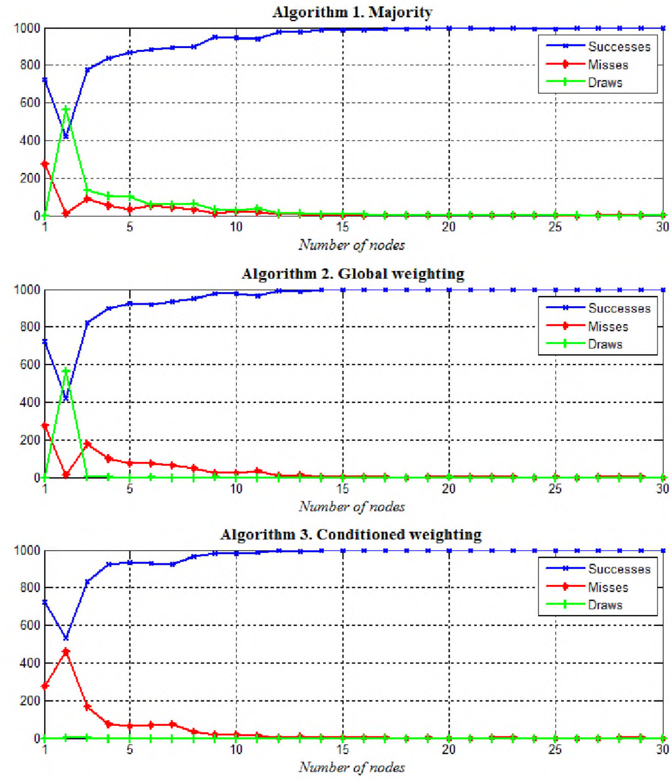

Fig. 8. First analysis: Number of successes, misses and draws for different network sizes.

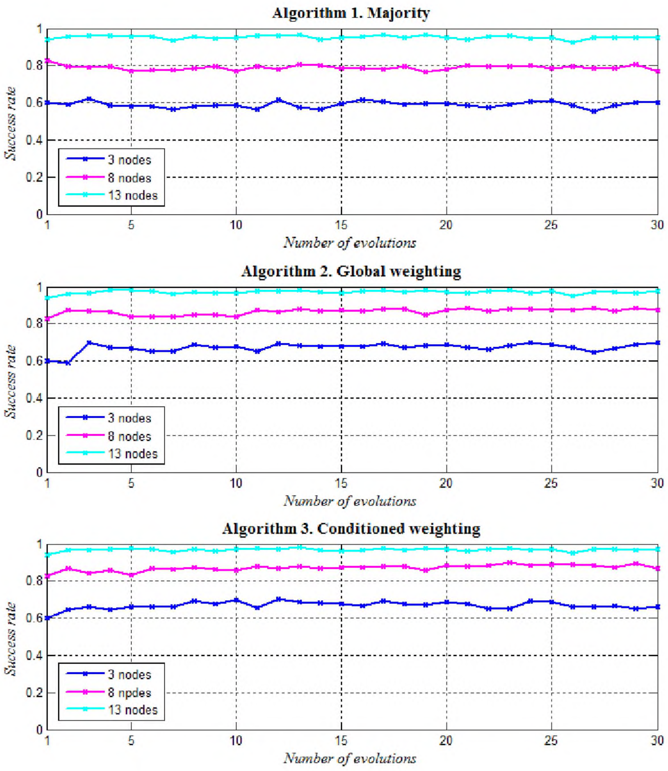

Fig. 9. Second analysis: Success rate versus number of evolutions for different network sizes.

TABLE II. MAIN CHARACTERISTICS OF SECOND ANALYSIS

	Alg. 1 Majority			*Alg.2 Global weighting*			*Alg.3 Conditioned weighting*		
Number of nodes	3	8	13	3	8	13	3	8	13
Mean	0,590	0,789	0,952	0,672	0,865	0,971	0,670	0,871	0,968
Slope	0,02%	-0,18%	0,04%	0,33%	0,16%	0,12%	0,22%	0,13%	0,11%

Finally, the third analysis seeks the relation between the hit rate and the number of evolutions considering different values for the learning rate and maintaining a constant network size. The analysis shown in Fig. 10 employs 8 nodes with a success rate ranging between 40% and 60% that is strongly lower than the used in previous analysis. It demonstrates that degradation appears in case of using conditioned weighting procedure. However, if the hit rates of nodes are increased, allowing rates up to 90%, this degradation is masked by the nodes with higher likelihood of success. Moreover, if the network size is reduced the behavior of the entire system does not change a lot. The difference lies in how the system evolves because using only 3 nodes the variations between evolutions are more abrupt.

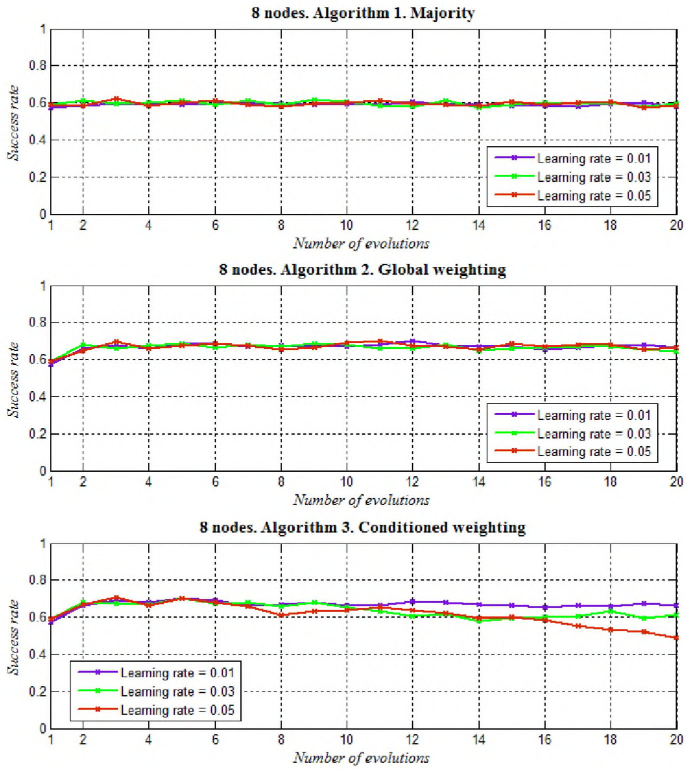

Fig. 10. Third analysis: Success rate versus number of evolutions for different learning rates.

VI. CONCLUSIONS

The most remarkable conclusion is that learning strategies allow to improve the system behavior considering past experiences as demonstrated in the first and second analysis. The first one proves that the system unawareness is higher using the algorithm number 1 because the algorithms 2 and 3 reduce the number of draws for every identification process. Moreover, the second analysis shows that the slopes of the algorithm 1 are negligible since they are almost zero. It means that the majority voting procedure does not improve over time. However, the slopes related to the other algorithms demonstrate that weighting procedures are able to enhance the success rate as more objects are detected. Comparing the two weighted voting procedures, in terms of learning, both have the same trend as demonstrated in the second analysis. However, in some cases algorithm 3, based on conditioned weighting, undergoes degradation as shown in the results of the third analysis. As the learning rate increases, this degradation also rises. Moreover, global weighting procedure, implemented by the algorithm number 2, is more stable, less computationally expensive and requires fewer resources.

In spite of the obtained results which demonstrate that the use of global weights is the best alternative, the conditioned weighting procedure should provide better behavior. However, it does not occur and the reason may be the lack of characterization of the environment model. Therefore, future works are focused on improving the system behavior and modeling the environment considering more features. Despite of this undesirable behavior, using the global weighting-based voting a significant improvement has been achieved in terms of success rates. Employing the initial system, which does not implement any cooperative model, the best success rates were established around 80%. However, the new system, applying cooperative learning strategies, is able to reach more than 90% of successes using networks of an acceptable size. Less than 5 nodes are required to provide a number of successes that overcome 90%. Additionally, if the nodes are placed properly its own behavior will be better and the success rate will be increment up to almost 99.99%.

As a main conclusion and to the aim of improving the conditioned weighting procedure, in future approaches a more adaptive model will be created in order to extend the proposed cooperative one to other applications and other kind of environments.

ACKNOWLEDGMENT

Authors wish to express their deep gratitude to *Luix Iluminación Inteligente S.L.* for their collaboration and support.

The previous work related to public lighting application has been developed under the call INNPACTO-2011 (ITP-2011-1977-920000), financed by the Spanish Ministry of Economy and Competitiveness (previously Ministry of Science and Innovation) within the National Plan for Scientific Research, Development and Technological Innovation, 2008-2011.